Local performance and margins in fuzzy classification trees

Alberto Suárez (*) and James F. Lutsko (**)

***Abstract*-- A fuzzy decision tree partitions the space of predictor variables into regions that are not disjoint. Instances to be classified are assigned to all these regions with different degrees of membership. The final class assignment is made by means of a voting scheme involving all the terminal nodes of the tree. Besides some improvements in the global classification error rate, the main benefit derived from the representation given by a fuzzy tree is the possibility of quantifying the classification margin for a given test example. The margin is obtained in terms of the fuzzy entropy, which measures how definite is the class assignment produced by the voting scheme. This procedure allows the identification of regions in attribute space for which the error rate of the tree is significantly lower than the average error rate.**

***Index terms*-- Automatic induction, decision trees, local performace, classification margins.**

I. INTRODUCTION

In a classification problem, a classical crisp decision tree classifies examples characterized by a vector of predictor variables (attributes) by dividing of the space of attributes into distinct areas where the class assignment can be made with a greater degree of certainty [1,2]. These areas are determined by a succession of Boolean tests. Each of these tests splits the data into two disjoint sets. The tree can be viewed as a hierarchy of Boolean tests whose parameters are determined by a succession of local optimizations of a specified quality function (e.g. the decrease of the impurity function [1], or the information gain [2] brought about by the split). It is expected that this greedy algorithm find a solution that is close to being globally optimal.

One of the disadvantages of the knowledge representation given by a crisp classification tree is that the notion of locality is partially lost. In particular, two examples that are very close to each other in attribute space, but on opposite sides of a split defined by a Boolean test are classified by separate branches of the tree. In order to regain the notion of locality and to be able to define a training algorithm that involves the minimization of a global cost function, a fuzzy decision tree can be constructed by replacing the Boolean tests in the internal nodes of the tree by fuzzy tests whose outcome are real-valued degrees of membership [3]. This means that points are split into sets that are not disjoint;

i.e. points may have a partial degree of membership to each of the sets that are defined by the split. A first stage of the classification process involves finding the memberships of the test example in the leaves. Once these values are computed, they are used to determine the final class assignment by a weighted voting procedure that uses the class labels produced by each of the leaves.

An additional advantage of the fuzzification algorithm introduced in [3] is that it allows the quantification of the margins associated to the classification of a given example. In the recent literature, the question of classification margins has been recognized as a key element in the performance of some families of classifiers. The classification margin can be intuitively defined as the certainty with which a classification is made. Vapnik [4] exploits this concept in the construction optimal margin classifiers, which use the inductive principle that the best classifiers are those with large minimal margins. In particular, the following procedure is suggested: The original classification problem is embedded in a high dimensional space where the examples in the training set are linearly separable. The margin is then defined in terms of the minimal distance to the separating hyperplane for points of different classes. The optimal separating hyperplane is the one that maximizes the minimal margin. One can also specify a similar algorithm and generalize the concept of margins for the case where the problem is not linearly separable even in the augmented space [4].

In the context of classification with ensembles of decision trees [5-8], the concept of margins has been also invoked to account for the performance of boosting methods [8]. The objective of boosting [7] is to generate an ensemble of classifiers, where the training examples that are more difficult to classify are progressively given a higher relevance in the construction of the "boosted" trees. The classification is made by a weighted voting procedure amongst the classifiers in the ensemble. Margins are then defined in terms of the preponderance of the majority class in the voting scheme [8].

In a fuzzy decision tree, owing to the fact that all tree leaves are involved in the classification produced by the tree, it is not necessary to generate an ensemble of classifiers to be able to define margins. In a certain sense, the conflicting class assignments given by the different leaves of the tree correspond to the conflicting votes of individuals in an ensemble of trees. Hence, a relative measure for the margin of the classification of an example can be given in terms of how diffuse its classification is.

(*) ETS de Informática, Universidad Autónoma de Madrid. Campus de Canto Blanco, 28049 Madrid, Spain. E-mail: alberto.suarez@ii.uam.es.

(**) Center for Nonlinear Phenomena and Complex Systems, Université Libre de Bruxelles, 1050-Brussels, Belgium.

The correlation between small margins and high inaccuracy in the classification can be understood in the light of the following observation: Those points whose classification involves several terminal nodes with conflicting class assignments are in a region of attribute space where a relevant split is being made. These regions are in general zones where there is a non zero probability of finding examples from different classes, either because of overlapping class probability distributions, or due to the presence of noise. In these regions the classification problem is intrinsically more difficult. On the other hand, when, in the classification of an example by a fuzzy decision tree, the class assignment is made by a single leaf node, or by a set of leaf nodes which produce the same class prediction, one can be more confident of the accuracy of the classification.

This observation has important consequences in gaining understanding of the distribution of errors in attribute space. The information could be used in various ways: One can use it to identify regions with an error rate much lower or much higher than the average rate. The classification of instances with a low fuzzy entropy will be accepted as more accurate. On the other hand, examples classified with a high fuzzy entropy can be handled separately, either by a human expert, or by another classifier whose bias is in a certain sense "complementary" to the bias of the decision tree (i.e. the second classifier is better at classifying those examples for which the decision tree does not exhibit a good performance). Thus, a heuristic based on the fuzzy entropy to combine the operation of different classifiers in order to improve the overall classification performance, can be developed.

II. FUZZY CLASSIFICATION TREES

In this section we review the algorithm presented in [3] to generate fuzzy classification trees by automatic induction from a set of data. Each of the examples in the training set consists of an ordered pair $(\mathbf{x}_n, y_n)$, where the first component is the D-dimensional vector of predictor variables, and the second component is the classification label of the example. Each of the examples belongs to one of K different classes. The starting point used in [3] is a CART decision tree [1]. In the CART algorithm, the classification tree is grown using the Gini impurity criterion to select the locally optimal splits at each stage. The fully grown tree is later pruned to its near-optimal size by minimizing a cost-complexity function. The resulting binary tree can be thought of as a hierarchy of Boolean tests on the predictor variables. This series of tests associate instances characterized by their vector of attributes, $\mathbf{x}$, to the nodes making up the tree. In fact, a node in the tree can be identified with the subset of instances assigned to it by the hierarchical questionnaire. It is thus possible to characterize node t by a membership function $\mu_t(\mathbf{x})$, which expresses in an alternative way the assignment of the examples: A value of the membership function $\mu_t(\mathbf{x}_n)=1$ indicates that the instance characterized by the vector of attributes $\mathbf{x}_n$ is assigned to node t by the hierarchy of Boolean tests. Conversely, a value 0 for the membership function indicates that example $\mathbf{x}_n$ is not assigned to that node by the tests. At the top of the hierarchy is the root node, which has all examples assigned to it. In terms of the degree of membership function, we have

$$\mu_{root}(\mathbf{x}_n)=1, \quad \forall\, \mathbf{x}_n . \tag{2.1}$$

Let us focus on the inner node t_i. Asuming only ordinal attributes are present, this node is split by the numerical Boolean test

$$\mathbf{c}_i \cdot \mathbf{x}_n > a_i \tag{2.2}$$

into two nodes. Examples for which Eq. (2.2) is true are assigned to the left child node, t_{iL}. The remainder are assigned to the right child node, t_{iR}.

The split at this inner node can be expressed in terms of membership functions in the child nodes

$$\begin{aligned} \mu_{iL}(\mathbf{x}) &= \mu_i(\mathbf{x})\ \theta(\mathbf{c}_i \cdot \mathbf{x}_n - a_i) \\ \mu_{iR}(\mathbf{x}) &= \mu_i(\mathbf{x})\ \theta(a_i - \mathbf{c}_i \cdot \mathbf{x}_n) \end{aligned}, \tag{2.3}$$

with the definition

$$\theta(z)=\begin{cases} 1 & if\ z>0 \\ 1/2 & if\ z=0 \\ 0 & if\ z<0 \end{cases} . \tag{2.4}$$

The non-Boolean value of 1/2 generally does not obtain in practice. The membership functions in (2.3) reflect the assignment of examples to the child nodes: A membership of 1 in one child means the example is assigned to that node; by construction, the membership to the other child node is zero.

The fuzzification procedure proposed in [3] consists in replacing the crisp splits defined by (2.3) in the CART tree by fuzzy sigmoidal splits of the form

$$\begin{aligned} \mu_{iL}(\mathbf{x}) &= \mu_i(\mathbf{x})\ \mathbf{S}\big(-b_i(\mathbf{c}_i \cdot \mathbf{x}_n - a_i)\big) \\ \mu_{iR}(\mathbf{x}) &= \mu_i(\mathbf{x})\ \mathbf{S}\big(b_i(\mathbf{c}_i \cdot \mathbf{x}_n - a_i)\big) \end{aligned}, \tag{2.5}$$

with the definition

$$\mathbf{S}(z)=\frac{1}{1+\exp z} . \tag{2.6}$$

With this substitution, examples are assigned to both child nodes with a real-valued (i.e. no longer Boolean) degree of membership. It is therefore natural to identify the nodes in the tree with a fuzzy set [9,10].

The sigmoidal fuzzy split has one extra parameter with respect to the Boolean or crisp split. This extra parameter, b_i, can be thought of as the inverse width of the splitting region. This region is the band around the splitting threshold where examples are assigned a significant membership in both child nodes. As this inverse width parameter tends to infinity (i.e. the width of the splitting band goes to zero), the crisp split is recovered from the fuzzy sigmoidal split

$$\mathbf{S}\left(b_i(\mathbf{c}_i \cdot \mathbf{x}_n - a_i)\right) \xrightarrow[b_i \to \infty]{} \theta(\mathbf{c}_i \cdot \mathbf{x}_n - a_i). \quad (2.7)$$

The parameters of the fuzzy splits are determined by a global optimization algorithm, which has been specified in [3].

Once the parameters of the fuzzy splits have been determined, we can compute $\{\mu_l(\mathbf{x})\}_{l \in \tilde{T}}$, the real-valued degrees of membership of example $\mathbf{x}$ in the set of terminal nodes of the tree, $\tilde{T}$. The label $\bar{y}_l$ is the class prediction given by terminal node t_l. The classification procedure consists of three steps

- For all terminal nodes of the tree, $t_l \in \tilde{T}$, compute $\mu_l(\mathbf{x})$, the degree of membership of the unclassified example $\mathbf{x}$ to the leaf node t_l, by recursive application of (2.5).

- Calculate the total weight assigned to each of the classes by the classification tree

$$\mu^{(k)}(\mathbf{x}) = \sum_{l \in \tilde{T}} \mu_l(\mathbf{x})\, \delta(\bar{y}_l, k), \quad k = 1,2,\ldots K \quad (2.8)$$

- The final class label is obtained by the rule

$$class(\mathbf{x}) = max_k \left(\mu^{(k)}(\mathbf{x})\right) \quad (2.9)$$

Besides information about the class label, the set of weights associated to each class contains information about the classification margins for the example in question. The correlation between margins and accuracy of the classification is discussed in the following section.

III. Classification error, fuzzy entropy and margins

Consider solving a classification problem with *K* classes by generating a fuzzy decision tree with the help of the algorithm described in section II. The fuzzy nature of the tests implies that all examples are assigned a non zero probability to all nodes of the decision tree. This implies that the classification of a given instance in a fuzzy tree is made jointly by $\tilde{T}$, the set of terminal nodes of the decision tree. The class assignments of the different terminal nodes for a given example will generally be in conflict. The more disparate these classification labels are, the fuzzier the classification for that example will be. The extent of this effect can be quantified by defining the fuzzy entropy of an example $\mathbf{x}$

$$S_{Fuzz}(\mathbf{x}) = -\sum_{k=1}^{K} \mu^{(k)}(\mathbf{x})\,(1 - \mu^{(k)}(\mathbf{x})) \quad (3.1)$$

Large values of the fuzzy entropy indicate that the classification is ambiguous, possibly erroneous, because alternative class labels, other than the one selected by rule (2.9), are given a significant weight. Expression (3.1) is used to rank order the examples according to how fuzzy their classification by this particular tree is, and therefore, how uncertain the classification given by the tree is. Other choices of the fuzzy entropy function (e.g. one involving logarithms) are possible. They lead to similar ordering in the examples. Hence, the observation that there is a strong correlation between the degree of fuzziness of classification of an example, as measured by this rank ordering, and its misclassification rate does not depend on the particular functional form chosen for the fuzzy entropy function.

In summary, the fuzzy sigmoidal splits that are used in a fuzzy classification tree introduce a natural measure of the proximity of a given instance in the space of attributes to the relevant splits. Proximity to a split indicates that the example is located in a region where classes may coexist, which in turn implies that it is intrinsically more difficult for the tree to classify those instances accurately. Consequently, points with higher fuzzy entropy have a smaller classification margin and are more likely to be misclassified. Ranking a set of examples in terms of their fuzzy entropy gives us a definition for their relative margins and permits us to identify those examples for which the classification error should be smaller than the average error. The presence of this correlation and its consequences will be explored in the following section, devoted to experiments.

IV. Experiments

In order to test the efficiency of fuzzy classification trees and to validate the hypothesized relation between fuzzy entropy, margins and accuracy of classification, we perform a series of experiments on three different data sets.

The first collection of experiments is carried out on the Wisconsin Breast Cancer database [11]. In order to assess the quality of the classification given by a fuzzy decision tree N-fold cross-validation is used: The data is randomly divided in N groups, with the precaution that the class proportions are maintained approximately equal in all sets. We then select one of the groups and use it as a test set for the classifier generated from the remaining (N-1) groups. The procedure is then repeated N times with each of the cross-validation sets. In the experiments realized, N is equal to 10. The values reported in Table I are averages over these 10 runs, with the standard deviation reported between parentheses.

The first column of Table I displays the average classification error achieved by a CART decision tree. The second and third columns present the results for a fuzzified CART tree generated according to the prescription given in [3]. The optimization algorithm that fixes the parameters of the fuzzy tree is run with different values for the characteristic width of the fuzzy splits (see [3] for details). In the second column we report the results of the optimization selected by using solely the training data, with the algorithm described in [3]. The third column displays the results of the best of the optimizations, as measured by the percent error on the test set. This latter error is reported to assess the efficiency of the heuristic procedure specified in [3] to select the best split width values for the optimization. We observe that, although there is some improvement of the classification error by the fuzzification procedure, the results are far from optimal, and further improvement in the selection procedure of the best optimization result is needed.

The first line in the table corresponds to the original tree. The second line of Table I displays the results

obtained with a boosted tree, where extra weight has been given to examples which have smaller margins, as measured by their fuzzy entropy ranking. We observe that whereas some small improvement of the classification can be observed in the boosted crisp CART tree, the performance of the optimized fuzzy classification tree slightly deteriorates. A close examination of the fuzzy trees generated in both cases shows that the examples

	CART	FUZZY CART	FUZZY CART*
ORIGINAL TREE	0.070 (0.038)	0.040 (0.034)	0.024 (0.030)
BOOSTED TREE	0.064 (0.036)	0.043 (0.037)	0.029 (0.029)

TABLE I: Classification error for the Wisconsin breast cancer database [11].

SET	Error CART	Error Fuzzy CART	Error 1st quartile	Error 2nd quartile	Error 3rd quartile	Error 4th quartile
WAVEFORM	0.309 (0.026)	0.175 (0.020)	0.044 (0.014)	0.103 (0.029)	0.214 (0.044)	0.336 (0.057)
DIABETES	0.257 (0.020)	0.252 (0.013)	0.072 (0.048)	0.194 (0.061)	0.303 (0.051)	0.439 (0.045)

TABLE II: Classification results for the waveform and the diabetes databases.

misclassified tend to be the same ones. This observation suggests that boosting will not improve the performance of fuzzy classification trees, since one of the elements necessary to the effectiveness of boosting methods is that the tree become unstable as the weight of misclassified training examples is increased. We conjecture that the robustness of the classification given by the fuzzy tree will almost certainly hinder the effectiveness of methods such as bagging o boosting [5-8], which rely on the instability of the decision trees.

A second set of experiments has been carried out for the waveform (see Ref. [1] for a detailed description of how to construct this synthetic data set) and the diabetes data sets (UC Irvine Machine Learning Database Repository [11]). The results reported are averages over random partitions of the available data into separate test and training sets. The standard deviation as estimated from the experiments is given between parentheses. For the diabetes problem the training set consists of 500 examples, and the test set of the remaining 268. The waveform set is a synthetic data set proposed by Breiman et al. [1]. We have used 300 examples for training and 5000 examples for testing. The results presented in Table II (taken form Ref. [3]) illustrate the correlation between the fuzzy entropy and the classification error: The examples are first ordered according to their fuzzy entropy and subsequently, the error for each of the quartiles in the test data is reported. It is apparent that the lower quartiles exhibit a significantly smaller error rate than the higher ones.

A different form of exhibiting this correlation is to plot the cumulative error (measured as the fraction of the total error) against the rank of points ordered according to their fuzzy entropy. Under the hypothesis that the error likelihood is not correlated with the fuzzy entropy of an example, one should obtain approximately a straight line with a unit slope. The slope of the actual curve at each point multiplied by the global error rate is an estimate of the local error rate. The results presented in Figs. 1 and 2 for the diabetes and the waveform databases, respectively, show that both in the test and the training set the accumulated error curve has a characteristic concave form:

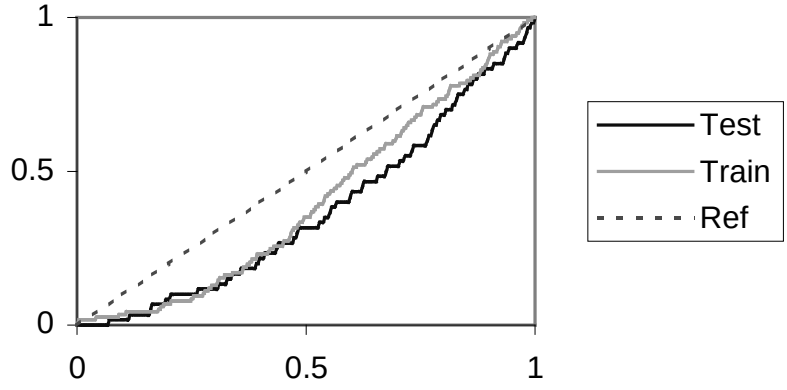

Figure 1. Accumulated error vs. rank of points ordered according to their fuzzy entropy for the diabetes data set.

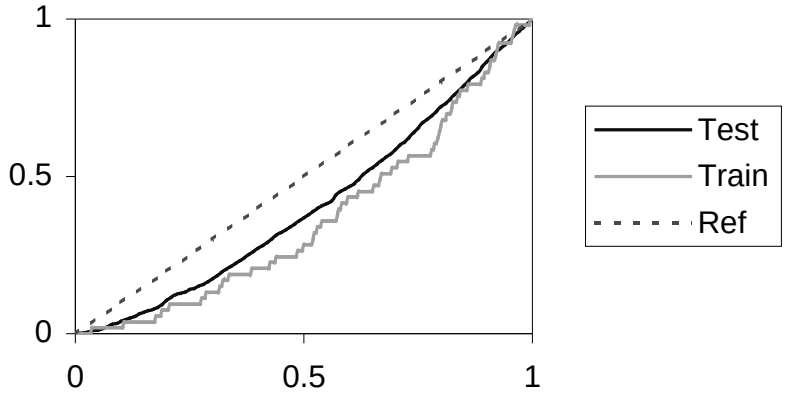

Figure 2. Accumulated error vs. rank of points ordered according to their fuzzy entropy for the waveform data set.

The curve starts off with a slope significantly lower than one. The slope gradually increases and eventually becomes larger than one as the abscissa approaches unity. These features indicate that examples with low fuzzy entropy have an error rate smaller than the average one. By contrast, examples with high values for the fuzzy entropy have a greater than average error rate.

V. CONCLUSIONS

The classification of a given instance in a fuzzy decision tree is a global process that involves all terminal nodes of the tree. In fact, the hierarchy of fuzzy tests produces a non-zero value for the degree of membership in all the leaves of the tree. Each of these terminal nodes predicts a different class label to the examples associated to it. Hence, the class has to be determined by a voting method, where the vote of a leaf node is weighted by the degree of membership of the example in that node. This voting procedure allows for a definition of classification margins in terms of the fuzzy entropy. This quantity is a measure of how clear is the vote in favor of a class and therefore of how confident we are on its classification. The conjecture that small classification margins (large fuzzy entropy) are correlated with the likelihood of misclassification is corroborated in a series of experiments on different standard databases. If the instances are ordered according to their fuzzy entropy, and then we plot the accumulated error vs. the rank of these points, the curve exhibits a characteristic concave shape. The slope of this curve, which rises from values smaller than one (points with lower fuzzy entropy) to values larger than one (points with higher fuzzy entropy), is a measure of the local error rate relative to the average error rate.

Although more experiments are needed, this work suggests that the information provided by the estimates of the margins does not seem to be useful in order to design a boosting algorithm similar to AdaBoost [7]. A probable reason for this behavior is the robustness of the classification given by fuzzy decision tree, which implies that the misclassified points are usually the same ones. They all tend to have high fuzzy entropy irrespective of the particular tree architecture generated by the fuzzy-CART algorithm from the training data.

The experiments presented in this work prompt in a number of directions. In particular, further research is needed to address the following problems: There seems to be some room for improvement in selection of the optimized tree in the algorithm proposed in [3]. It would also be desirable to have a quantitatively reliable relation between the fuzzy entropy, or local margin, and an actual measure of the local error rate. Finally, heuristics that exploit the information contained in the classification margins in order to improve the overall classification performance ought to be developed.

VI. ACKNOWLEDGEMENTS

A. S. gratefully acknowledges financial support from CICyT (Spain), project no. TIC98-0247-C02-02.

VII. BIBLIOGRAPHY

[1] L. Breiman, J. H. Friedman, R. A. Olshen and C. J. Stone, *Classification and Regression Tree*, Chapmann & Hall, New York, 1984

[2] J. R. Quinlan, *C4.5: Programs for Machine Learning*, Morgan-Kauffmann, San Mateo, 1993

[3] A. Suárez and J. F. Lutsko, "Globally Optimal Fuzzy Decision Trees for Classification and Regression," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 12, pp. 1297-1311, 1999

[4] V. N. Vapnik, *Statistical Learning Theory*, Wiley & Sons, New York, 1998

[5] L. Breiman, " Arcing Classifiers," *Annals of Statistics*, vol.26 no. 3, pp. 801-849, 1998

[6] T. G. Dietterich, "An experimental comparison of three methods for constructing ensembles of decision trees: bagging, boosting, and randomization," *Machine Learning*, vol. 40 no. 2 pp. 139-158, 2000.

[7] Y. Freund and R. E. Shapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of Computer and System Sciences*, vol. 55, pp. 119-139, 1997.

[8] R. E. Schapire, Y. Freund, P. Bartlett and W. S. Lee, "Boosting the margin: A new explanation for the effectiveness of voting methods," *Annals of Statistics*, vol. 26 no. 6, pp. 1651-1686, 1998.

[9] L. Zadeh, "Fuzzy Sets," *Information and Control*, vol.8, pp. 338-353, 1965.

[10] L. Zadeh, "Outline of a new approach to the analysis of Complex Systems and decision processes," *IEEE Trans. Systems, Man and Cybernetics*, vol.3, pp.28-44, 1973.

[11] C. L. Blake and C. J. Merz. "UCI Repository of machine learning databases," Department of Information and Computer

Science, University of California, Irvine, CA, 1998 [http://www.ics.uci.edu/~mlearn/MLRepository.html].